(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号

特開平10－301772

(43)公開日　平成10年(1998)11月13日

(51)Int.Cl.[6]	識別記号	Ｆ Ｉ	
Ｇ０６Ｆ　9/06	５５０	Ｇ０６Ｆ　9/06	５５０Ａ
			５５０Ｅ
			５５０Ｚ

審査請求　未請求　請求項の数14　ＯＬ　（全 15 頁）

(21)出願番号　特願平9－112179

(22)出願日　平成９年(1997)４月30日

(71)出願人　000002185
ソニー株式会社
東京都品川区北品川６丁目７番35号

(72)発明者　竹内　彰一
東京都品川区北品川６丁目７番35号　ソニー株式会社内

(72)発明者　雛波　慎二
東京都品川区北品川６丁目７番35号　ソニー株式会社内

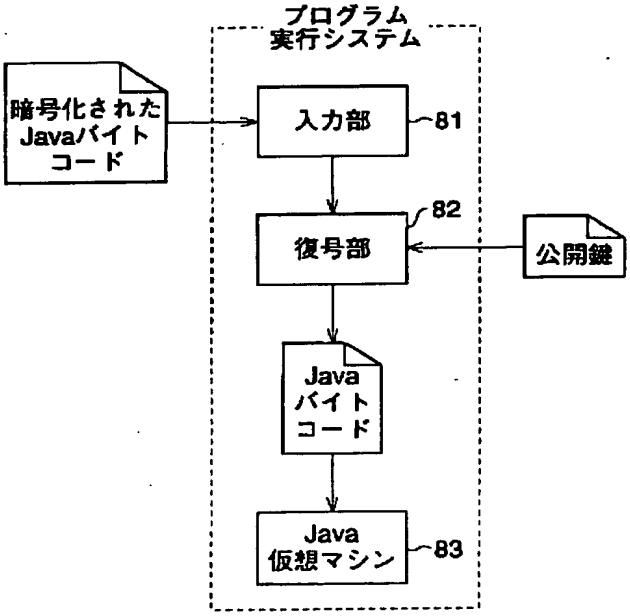
(74)代理人　弁理士　稲本　義雄

(54)【発明の名称】　情報処理装置および情報処理方法、並びに記録媒体

(57)【要約】

【課題】　あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにし、その他のプログラムの実行を制限する。

【解決手段】　プログラム実行システムを構成する入力部８１には、所定の秘密キーを用いて暗号化されたアプリケーションプログラムが入力され、その暗号化されたアプリケーションプログラムとしての暗号文は、そのまま、復号部８２に供給される。復号部８２には、入力部８１から暗号文が供給される他、その暗号文を作成するのに用いた秘密キーに対応する公開キーも供給されるようになされており、そこでは、公開キーを用いて、暗号文が復号化され、その復号結果としてのＪａｖａバイトコードが、Ｊａｖａ仮想マシン８３に供給される。Ｊａｖａ仮想マシン８３では、復号部８２からのＪａｖａバイトコードが解釈、実行される。



【特許請求の範囲】
【請求項１】　プログラムを実行するための処理を行う情報処理装置であって、
暗号化された前記プログラムを復号化する復号化手段と、
前記復号化手段が出力する前記プログラムを実行する実行手段とを備えることを特徴とする情報処理装置。
【請求項２】　プログラムを実行するための処理を行う情報処理方法であって、
暗号化された前記プログラムを復号化し、
その復号化されたプログラムを実行することを特徴とする情報処理方法。
【請求項３】　コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。
【請求項４】　暗号化されたプログラムの復号化が所定のキーを用いて行われる場合において、
前記所定のキーも記録されていることを特徴とする請求項３に記載の記録媒体。
【請求項５】　プログラムを処理する情報処理装置であって、
請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする情報処理装置。
【請求項６】　プログラムを処理する情報処理方法であって、
請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化することを特徴とする情報処理方法。
【請求項７】　請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする記録媒体。
【請求項８】　プログラムを実行するための処理を行う情報処理装置であって、
前記プログラムを実行する実行手段と、
前記プログラムが正当なものかどうかを確認する確認手段と、
前記確認手段により正当なものであることが確認された前記プログラムを、前記実行手段に供給する供給手段とを備えることを特徴とする情報処理装置。
【請求項９】　前記プログラムには、署名が付加されており、
前記確認手段は、前記署名に基づいて、前記プログラムが正当なものかどうかを確認することを特徴とする請求項８に記載の情報処理装置。
【請求項１０】　プログラムを実行するための処理を行う情報処理方法であって、
前記プログラムが正当なものかどうかを確認し、
正当なものであることが確認された場合のみ、前記プログラムを実行することを特徴とする情報処理方法。
【請求項１１】　コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、前記プログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。
【請求項１２】　プログラムを処理する情報処理装置であって、
請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする情報処理装置。
【請求項１３】　プログラムを処理する情報処理方法であって、
請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理することを特徴とする情報処理方法。
【請求項１４】　請求項８に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする記録媒体。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、情報処理装置および情報処理方法、並びに記録媒体に関し、特に、例えば、あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにする情報処理装置および情報処理方法、並びに記録媒体に関する。
【０００２】
【従来の技術】最近急速に普及してきたインターネットに適していることから、Ｊａｖａ（米国Sun Microsystems社の商標）が注目されている。Ｊａｖａは、オブジェクト指向言語であるＪａｖａ言語や、そのＪａｖａ言語で記述されたプログラム（以下、適宜、Ｊａｖａプログラムという）の実行に適したプロセッサのアーキテクチャを定義した仮想マシン（以下、適宜、Ｊａｖａ仮想マシンという）、その他のＪａｖａと関連する要素を呼ぶのに、あるいは、それらの総称として用いられる。なお、ここでいう仮想マシンとは、ユーザに対して１台のコンピュータを複数のコンピュータに仮想的に見せる場合の仮想マシンではなく、言語処理系を実装する場合に想定する仮想的なマシンを意味する。
【０００３】Ｊａｖａ仮想マシンは、種々のソフトウェアやＯＳ（Operating System）、ハードウェアで動作するように実装される。一方、Ｊａｖａプログラムは、Ｊａｖａ仮想マシンの命令セットからなるバイナリコードにコンパイルされる。このバイナリコードは、Ｊａｖａ仮想マシンが動作するどのようなハードウェアでも実行することができる。従って、Ｊａｖａ仮想マシンさせ動作すれば、コンパイル済みのＪａｖａプログラムは、種々のプラットフォームで実行することができる。

【０００４】
【発明が解決しようとする課題】上述したように、Ｊａｖａ仮想マシンを実装すれば、どのようなマシン上でも、Ｊａｖａプログラムを実行することができること等から、Ｊａｖａ仮想マシンが、多数のユーザに普及することが予想され、さらに、そのような多数のユーザ向けに、多くのアプリケーションプログラムが開発、配布（有償、無償を問わない）されることが予想される。
【０００５】このような状況の下、Ｊａｖａ仮想マシンその他のプログラム実行環境を開発または配布した者からすれば、自身が開発等したプログラム実行環境下において実行される、第三者により開発されたアプリケーションプログラムの配布を制限したい場合がある。即ち、例えば、ライセンス契約を結んだ者だけに、アプリケーションプログラムの配布を許可したい場合がある。
【０００６】一方、Ｊａｖａ仮想マシンでは、ＪａｖａコンパイラでＪａｖａプログラムを、バイトコード（Ｊａｖａバイトコード）と呼ばれる中間コードにコンパイルしたものが解釈されて実行されるが、Ｊａｖａバイトコードは、それを逆コンパイルすることにより、比較的容易に理解することができるため、リバースエンジニアリングを簡単にすることができる。従って、アプリケーションプログラムの、他人による模倣や改竄を防止する必要がある。
【０００７】本発明は、このような状況に鑑みてなされたものであり、あるプログラム実行環境下におけるプログラムの実行を制限することができるようにし、さらに、プログラムの模倣や改竄を防止することができるようにもするものである。
【０００８】
【課題を解決するための手段】請求項１に記載の情報処理装置は、暗号化されたプログラムを復号化する復号化手段と、復号化手段が出力するプログラムを実行する実行手段とを備えることを特徴とする。
【０００９】請求項２に記載の情報処理方法は、暗号化されたプログラムを復号化し、その復号化されたプログラムを実行することを特徴とする。
【００１０】請求項３に記載の記録媒体は、コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されていることを特徴とする。
【００１１】請求項５に記載の情報処理装置は、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする。
【００１２】請求項６に記載の情報処理方法は、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化することを特徴とする。
【００１３】請求項７に記載の記録媒体は、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする。
【００１４】請求項８に記載の情報処理装置は、プログラムが正当なものかどうかを確認する確認手段と、確認手段により正当なものであることが確認されたプログラムを、そのプログラムを実行する実行手段に供給する供給手段とを備えることを特徴とする。
【００１５】請求項１０に記載の情報処理方法は、プログラムが正当なものかどうかを確認し、正当なものであることが確認された場合のみ、プログラムを実行することを特徴とする。
【００１６】請求項１１に記載の記録媒体は、コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されていることを特徴とする。
【００１７】請求項１２に記載の情報処理装置は、請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする。
【００１８】請求項１３に記載の情報処理方法は、請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理することを特徴とする。
【００１９】請求項１４に記載の記録媒体は、請求項８に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする。
【００２０】請求項１に記載の情報処理装置においては、復号化手段は、暗号化されたプログラムを復号化し、実行手段は、復号化手段が出力するプログラムを実行するようになされている。
【００２１】請求項２に記載の情報処理方法においては、暗号化されたプログラムを復号化し、その復号化されたプログラムを実行するようになされている。
【００２２】請求項３に記載の記録媒体には、コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。
【００２３】請求項５に記載の情報処理装置においては、暗号化手段が、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。
【００２４】請求項６に記載の情報処理方法においては、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。
【００２５】請求項７に記載の記録媒体には、請求項１に記載の情報処理装置で実行可能なコードに復号化され

る暗号文に、プログラムを暗号化したものが記録されている。

【００２６】請求項８に記載の情報処理装置においては、確認手段は、プログラムが正当なものかどうかを確認し、供給手段は、確認手段により正当なものであることが確認されたプログラムを、そのプログラムを実行する実行手段に供給するようになされている。

【００２７】請求項１０に記載の情報処理方法においては、プログラムが正当なものかどうかを確認し、正当なものであることが確認された場合のみ、プログラムを実行するようになされている。

【００２８】請求項１１に記載の記録媒体には、コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されている。

【００２９】請求項１２に記載の情報処理装置においては、処理手段が、請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理するようになされている。

【００３０】請求項１３に記載の情報処理方法においては、請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理するようになされている。

【００３１】請求項１４に記載の記録媒体には、請求項８に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されている。

【００３２】

【発明の実施の形態】以下に、本発明の実施の形態を説明するが、その前に、特許請求の範囲に記載の発明の各手段と以下の実施の形態との対応関係を明らかにするために、各手段の後の括弧内に、対応する実施の形態（但し、一例）を付加して、本発明の特徴を記述すると、次のようになる。

【００３３】即ち、請求項１に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であって、暗号化されたプログラムを復号化する復号化手段（例えば、図１２に示す復号部８２など）と、復号化手段が出力するプログラムを実行する実行手段（例えば、図１２に示すＪａｖａ仮想マシン８３など）とを備えることを特徴とする。

【００３４】請求項５に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段（例えば、図１０に示すプログラムの処理ステップＳ１４など）を備えることを特徴とする。

【００３５】請求項８に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であって、プログラムを実行する実行手段（例えば、図１６に示すＪａｖａ仮想マシン８３など）と、プログラムが正当なものかどうかを確認する確認手段（例えば、図１６に示す署名確認部１０３など）と、確認手段により正当なものであることが確認されたプログラムを、実行手段に供給する供給手段（例えば、図１６に示す仮想マシン入力制御部１０４など）とを備えることを特徴とする。

【００３６】請求項１２に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理する処理手段（例えば、図１４に示すプログラムの処理ステップＳ２４およびＳ２５など）を備えることを特徴とする。

【００３７】なお、勿論この記載は、各手段を上記したものに限定することを意味するものではない。

【００３８】次に、本発明は、Ｊａｖａのような仮想マシンの他、現実のマシンそのものについても適用可能であるが、ここでは、本発明を、Ｊａｖａ仮想マシンに適用した場合を例に説明する。

【００３９】なお、Ｊａｖａについては、例えば、日経ＢＰ社発行の日経エレクトロニクスの１９９６．３．２５（ｎｏ．６５８）、同１９９６．６．１７（ｎｏ．６６４）などに、その詳細が記載されている。そこで、ここでは、簡単に説明する。

【００４０】Ｊａｖａ仮想マシンは、抽象化された実行機械であり、その実態は、実際の計算機で実行されるプログラムである。そして、Ｊａｖａ仮想マシンは、実際の計算機と同様に、プログラムカウンタや、スタックレジスタ、汎用レジスタ、スタックやヒープとしてのメモリ、その他の資源を有するが、これらの資源は、実際の計算機の資源にマッピングされている。

【００４１】即ち、図１に示すように、いま実際の計算機１が、中央演算処理装置２、その中央演算処理装置２が内蔵するレジスタ３、メモリ４などの資源を有するとして、この計算機１に、Ｊａｖａ仮想マシン１１を実装すると、そのＪａｖａ仮想マシン１１の資源として、実際の計算機１の資源がマッピングされる。図１の実施の形態では、Ｊａｖａ仮想マシン１１は、資源として、レジスタ１３や、メモリ１４などを有しており、レジスタ１３はレジスタ３に、メモリ１４の２００番地は、メモリ４の１００番地に、それぞれマッピングされている。

【００４２】実際の計算機１では、中央演算処理装置２に対する命令は、その資源に対する操作として実行されるが、Ｊａｖａ仮想マシン１１においても、その資源に対する操作として実行される命令が定義されている。このＪａｖａ仮想マシン１１に対する命令を記述するための言語がＪａｖａ言語であり、Ｊａｖａ仮想マシン１１では、このＪａｖａ言語で記述されたソースプログラムを、Ｊａｖａコンパイラで、Ｊａｖａバイトコードにコンパイルしたものが、解釈、実行される。

【００４３】即ち、図２に示すように、Ｊａｖａ言語で記述されたソースプログラムであるＪａｖａ言語プログラムは、Ｊａｖａコンパイラ２１でコンパイルされ、Ｊａｖａバイトコードとされる。このＪａｖａバイトコードが、Ｊａｖａ仮想マシン１１に入力され、Ｊａｖａ仮想マシン１１では、Ｊａｖａバイトコードが、実際の計算機１（中央演算処理装置２）が解釈することのできる機械語コード（マシン語）に変換される。具体的には、例えば、図３に示すように、「ｍｏｖｅ　＃１２５、レジスタ１３」というＪａｖａバイトコードで記述された「数字の１２５を、レジスタ１３にセット」という命令（Ｊａｖａバイトコード命令）が、Ｊａｖａ仮想マシン１１に入力された場合、Ｊａｖａ仮想マシン１１では、このＪａｖａバイトコードが、「ｍｏｖｅ　＃１２５，レジスタ３」という機械語コードで記述された命令（マシン語命令）に変換される。

【００４４】そして、計算機１では、この機械語コードで記述された命令が実行されることにより、図４（Ａ）に示すように、計算機１のレジスタ３に、数字の１２５がセットされる。

【００４５】上述したように、計算機１のレジスタ３には、Ｊａｖａ仮想マシン１１のレジスタ１３がマッピングされており、従って、図４（Ａ）に示したように、計算機１のレジスタ３に、数字の１２５をセットすることは、Ｊａｖａ仮想マシン１１から見れば、図４（Ｂ）に示すように、そのレジスタ１３に数字の１２５をセットすることになる。

【００４６】以上のように、Ｊａｖａ仮想マシン１１へのＪａｖａバイトコードによる命令は、計算機１用の機械語コードに変換され、Ｊａｖａ仮想マシン１１の資源にマッピングされている計算機１の資源に対する操作として実行される。この操作は、Ｊａｖａ仮想マシン１１から見れば、その資源（Ｊａｖａ仮想マシン１１の資源）に対する操作となり、Ｊａｖａバイトコードによる命令が実行されたことになる。

【００４７】従って、前述したように、Ｊａｖａ仮想マシンを、実際の計算機（コンピュータ）に実装することで、その計算機が使用しているＣＰＵ（Central ProcessingUnit）やＯＳに無関係に、コンパイル済みのＪａｖａプログラムを実行することができる。

【００４８】ここで、Ｊａｖａバイトコードを機械語コードに変換して実行する手法としては、例えば、Ｂａｓｉｃ言語によるプログラムを実行する場合に採用されている、各命令を機械語コードに逐次翻訳して実行するインタープリタ形式と、命令を一括して機械語コードに翻訳して実行するＪＩＴ（Just In Time）コンパイラ形式などがある。

【００４９】なお、Ｂａｓｉｃ言語によるプログラムを実行する場合に採用されているインタープリタ形式は、そのソースコードを翻訳する点で、中間コードであるＪａｖａバイトコードを翻訳する場合と異なるが、ここでは、これらを特に区別しない（区別する必要もない）。

【００５０】次に、図５は、本発明を適用したプログラム提供システム（システムとは、複数の装置が論理的に集合したものをいい、各構成の装置が同一筐体中にあるか否かは問わない）の一実施の形態の構成例を示している。

【００５１】このプログラム提供システムにおいては、ソフトウェア開発者が、プログラム認証機関により認証されていないアプリケーションプログラムをユーザに配布した場合に、そのユーザ端末３３上での、そのアプリケーションプログラムの実行を制限するようになされている。

【００５２】即ち、ソフトウェア開発者は、例えば、Ｊａｖａ仮想マシン上で動作するアプリケーションプログラムを開発すると、それをコンパイルしてＪａｖａバイトコードとしたものを、そのソフトウェア開発者サーバ３１から、例えば、インターネットや、公衆回線、ＣＡＴＶ網、地上波、衛星回線、その他でなるネットワーク３４を介して、プログラム認証機関サーバ３２に送信する。

【００５３】プログラム認証機関サーバ３２では、ソフトウェア開発者サーバ３１からＪａｖａバイトコードを受信すると、それを認証し、その認証したＪａｖａバイトコードを、ネットワーク３４を介して、ソフトウェア開発者サーバ３１に送信する。ソフトウェア開発者サーバ３１では、プログラム認証機関サーバ３２からの認証されたＪａｖａバイトコードであるアプリケーションプログラムが受信され、記憶される。

【００５４】そして、ソフトウェア開発者サーバ３１においては、ユーザ端末３３からアプリケーションプログラムの要求があると、そのアプリケーションプログラムを、ネットワーク３４を介して、ユーザ端末３３に送信する。ユーザ端末３３には、プログラム認証機関、またはプログラム認証機関にプログラムの認証を依頼している者が開発または配布したプログラム実行環境としての、例えば、Ｊａｖａ仮想マシンが実装されている。そして、Ｊａｖａ仮想マシンとしてのユーザ端末３３では、ソフトウェア開発者サーバ３１から受信したアプリケーションプログラムが、プログラム認証機関によって認証されているものであるときのみ、そのアプリケーションプログラムが、正常に実行される。

【００５５】即ち、ソフトウェア開発者サーバ３１から受信したアプリケーションプログラムが、プログラム認証機関によって認証されていない場合には、Ｊａｖａ仮想マシンとしてのユーザ端末３３では、そのアプリケーションプログラムは実行されない（実行されても、正常には実行されない）。

【００５６】従って、結果として、プログラム実行環境としてのＪａｖａ仮想マシンを開発等した者は、そのＪ

ａｖａ仮想マシンにおいて実行される、第三者により開発されたアプリケーションプログラムの、いわば勝手な配布を制限し、例えば、ライセンス契約を結んだソフトウェア開発者だけに、アプリケーションプログラムの配布を許可することができる。

【００５７】なお、ソフトウェア開発者が、アプリケーションプログラムを、例えば、ＣＤ（Compact Disc）－ＲＯＭ、磁気ディスク、その他でなる記録媒体３５に記録して、郵送、店頭における販売、その他の手段で、ユーザに配布する場合においても、上述の場合と同様に、そのアプリケーションプログラムが、プログラム認証機関によって認証されていないときには、Ｊａｖａ仮想マシンとしてのユーザ端末３３では、そのアプリケーションプログラムは実行されない。

【００５８】また、上述の場合においては、ソフトウェア開発者とプログラム認証機関との間で、ネットワーク３４を介して、データのやりとりを行うようにしたが、これらの間でのデータのやりとりは、そのデータを記録した記録媒体を郵送などすることによって行うことも可能である。

【００５９】さらに、図５の実施の形態では、ソフトウェア開発者サーバ３１、プログラム認証機関サーバ３２、またはユーザ端末３３を、それぞれ１つしか設けていないが、これらは、複数設けることが可能である。

【００６０】次に、図６は、図５のソフトウェア開発者サーバ３１の構成例を示している。

【００６１】ＣＰＵ４１は、補助記憶装置４６に記憶（記録）されたオペレーティングシステムの制御の下、同じく補助記憶装置４６に記憶されたプログラムを実行することで、各種の処理を行う。ＲＯＭ（Read Only Memory）４２は、例えば、ＩＰＬ（Initial Program Loading）のプログラムなどを記憶している。ＲＡＭ（Random Access Memory）４３は、ＣＰＵ４１が実行するプログラムや、ＣＰＵ１の動作上必要なデータを記憶する。入力部４４は、例えば、キーボードやマウスなどで構成され、所定のデータやコマンドを入力するときなどに操作される。出力部４５は、ディスプレイやプリンタなどで構成され、所定の情報を表示、印刷する。補助記憶装置４６は、例えば、ハードディスクなどで構成され、オペレーティングシステム、その他のＣＰＵ４１が実行するプログラムなどを記憶している。さらに、補助記憶装置４６は、ＣＰＵ４１の処理結果その他の必要なデータなども記憶する。通信制御部４７は、ネットワーク３４を介して行われる通信を制御する。

【００６２】次に、図７は、図５のプログラム認証機関サーバ３２の構成例を、図８は、図５のユーザ端末３３の構成例を、それぞれ示している。

【００６３】プログラム認証機関サーバ３２は、ＣＰＵ５１乃至通信制御部５７で構成され、また、ユーザ端末３３は、ＣＰＵ６１乃至通信制御部６７で構成され、これらは、図６におけるＣＰＵ４１乃至通信制御部４７とそれぞれ同様に構成されるので、その説明は省略する。

【００６４】次に、図９のフローチャートを参照して、ソフトウェア開発者サーバ３１の処理について説明する。

【００６５】ソフトウェア開発者が開発した、Ｊａｖａ仮想マシン上で実行されるアプリケーションプログラムは、例えば、補助記憶装置４６に記録（記憶）される。そして、ソフトウェア開発者サーバ３１では、まず最初に、ステップＳ１において、ＣＰＵ４１が、補助記憶装置４６に記憶されたアプリケーションプログラムを、Ｊａｖａコンパイラのプログラムにしたがってコンパイルし、Ｊａｖａバイトコードとする。このＪａｖａバイトコードは、再び、補助記憶装置４６に供給されて記憶される。

【００６６】そして、ステップＳ２に進み、通信制御部４７は、補助記憶装置４６に記憶されたアプリケーションプログラムとしてのＪａｖａバイトコードを読み出し、ネットワーク３４を介して、プログラム認証機関サーバ３２に送信し、ステップＳ３に進む。ステップＳ３では、ＣＰＵ４１において、認証されたアプリケーションプログラム（Ｊａｖａバイトコード）としての、そのアプリケーションプログラムを暗号化した暗号文が、プログラム認証機関サーバ３２から送信されてきたかどうかが判定され、まだ、送信されてきていないと判定された場合、ステップＳ３に戻る。

【００６７】また、ステップＳ３において、暗号文が送信されてきたと判定された場合、ステップＳ４に進み、通信制御部４７において、その暗号文が受信され、ステップＳ５に進む。ステップＳ５では、通信制御部４７で受信された暗号文が、補助記憶装置４６に転送されて記憶され、処理を終了する。

【００６８】次に、図１０のフローチャートを参照して、プログラム認証機関サーバ３２の処理について説明する。

【００６９】プログラム認証機関は、例えば、プログラム実行環境としてのＪａｖａ仮想マシンを開発または配布した者、あるいは、その者の依頼を受けた機関で、プログラム認証機関サーバ３２では、例えば、ライセンス契約を結んだ者からのアプリケーションプログラムを認証するプログラム認証処理を行うようになされている。

【００７０】即ち、プログラム認証機関サーバ３２のＣＰＵ５１では、まず最初に、ステップＳ１１において、例えば、ソフトウェア開発者サーバ３１などから、認証対象のアプリケーションプログラムとしてのＪａｖａバイトコードが、ネットワーク３４を介して送信されてきたかどうかが判定され、送信されてきていないと判定された場合、ステップＳ１１に戻る。また、ステップＳ１１において、Ｊａｖａバイトコードが送信されてきたと判定された場合、ステップＳ１２に進み、そのＪａｖａ

バイトコードが、例えば、ライセンス契約を結んだソフトウェア開発者（以下、適宜、正規のソフトウェア開発者という）からのものであるかどうかが、ＣＰＵ５１によって判定される。

【００７１】即ち、プログラム認証機関は、Ｊａｖａ仮想マシン上で実行されるアプリケーションプログラムの開発、販売等を許可することについてのライセンス契約を、ソフトウェア開発者との間で結ぶと、そのソフトウェア開発者に対して、例えば、ＩＤおよびパスワードを発行する。そして、このようなライセンス契約を結んだ正規のソフトウェア開発者からは、認証対象のＪａｖａバイトコードとともに、ライセンス契約時に発行されたＩＤおよびパスワードが送信されるようになされており、ステップＳ１２では、このＩＤおよびパスワードに基づいて、Ｊａｖａバイトコードが、正規のソフトウェア開発者からのものであるかどうかが判定される。

【００７２】ステップＳ１２において、Ｊａｖａバイトコードが、正規のソフトウェア開発者からのものでないと判定された場合、即ち、ライセンス契約を結んでいないソフトウェア開発者からＪａｖａバイトコードが送信されてきた場合、ステップＳ１３に進み、通信制御部５７において、そのソフトウェア開発者に対して、ライセンス契約を結ばなければ、Ｊａｖａバイトコードを認証することができない旨のメッセージが送信され、処理を終了する。

【００７３】一方、ステップＳ１２において、Ｊａｖａバイトコードが、正規のソフトウェア開発者からのものであると判定された場合、ステップＳ１４に進み、ＣＰＵ５１において、そのＪａｖａバイトコードが暗号化され、これにより暗号文とされることで、その認証が行われる。

【００７４】そして、ステップＳ１５に進み、通信制御部５７において、Ｊａｖａバイトコードの認証結果としての、その暗号文が、Ｊａｖａバイトコードを送信してきたソフトウェア開発者、即ち、ここでは、例えば、ソフトウェア開発者サーバ３１に、ネットワーク３４を介して送信され、処理を終了する。

【００７５】次に、プログラム認証機関サーバ３２において行われる、ステップＳ１４の暗号化の方法について説明する。

【００７６】プログラム認証機関サーバ３２では、Ｊａｖａバイトコードの暗号化が、例えば、ＲＳＡ方式（ＭＩＴの３人の研究者により考案されたもので、ＲＳＡは、その３人の研究者の頭文字をとったものである）などに代表される公開鍵暗号化方式によって行われるようになされている。

【００７７】即ち、図１１は、公開鍵暗号化方式による暗号化／復号化システムの構成例を示している。

【００７８】暗号化器７１には、暗号化対象である平文が入力される。そして、暗号化器７１では、秘密キーと呼ばれる各個人に特有の暗号キーを用いて、平文が暗号化され、暗号文とされる。

【００７９】一方、復号化器７２には、暗号化器７１で暗号化された暗号文が入力される。そして、復号化器７２では、公開キーと呼ばれる広く一般に公開されている復号キーを用いて、暗号文が復号化され、元の平文とされる。

【００８０】プログラム認証機関サーバ３２では、ソフトウェア開発者サーバ３１からのＪａｖａバイトコードが、それに特有の秘密キーを用いて暗号化されて、暗号文とされる。

【００８１】なお、暗号化の手法は、公開鍵暗号化方式に限定されるものではなく、その他、例えば、ＤＥＳ（Data Encryption Standard）方式（ＩＢＭ社により開発され、米国連邦政府の標準として実用化されたもの）などに代表される秘密鍵暗号方式その他の方式を採用することが可能である。

【００８２】次に、図１２は、ユーザ端末３３においてアプリケーションプログラムを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。

【００８３】入力部８１は、暗号文（暗号化されたＪａｖａバイトコード）を受け付け、復号部８２に供給するようになされている。復号部８２は、例えば、図１１の復号化器７２としての処理を行うもので、入力部８１の出力を、公開キーを用いて復号化し、元のＪａｖａバイトコードとするようになされている。復号部８２で得られたＪａｖａバイトコードは、Ｊａｖａ仮想マシン８３に供給されるようになされており、Ｊａｖａ仮想マシン８３は、復号部８２からのＪａｖａバイトコードにしたがった処理を実行するようになされている。

【００８４】以上のように構成されるプログラム実行システムでは、まず最初に、入力部８１において、アプリケーションプログラムとしての暗号文が取得される。即ち、例えば、あらかじめ、ソフトウェア開発者サーバ３１から、ネットワーク３４を介して、アプリケーションプログラムとしての暗号文を受信し、その暗号文が、ファイルとして、ユーザ端末３３の補助記憶装置６７に記憶されている場合や、アプリケーションプログラムとしての暗号文がファイルとして記録された記録媒体３５がユーザ端末３３にセットされる場合などには、入力部８１は、そのファイルをオープンし、暗号文を読み出す。

【００８５】また、例えば、ソフトウェア開発者サーバ３１が、インターネットとしてのネットワーク３４に接続されており、そのようなソフトウェア開発者サーバ３１において、アプリケーションプログラムとしての暗号文が、ＵＲＬ（Uniform Resource Locator）と対応付けられている場合に、ユーザが、入力部６４を操作して、そのＵＲＬを指定したときには、入力部８１は、ソフトウェア開発者サーバ３１からネットワーク３４を介して

送信されてくるアプリケーションプログラムとしての暗号文を受信する。

【００８６】さらに、例えば、ソフトウェア開発者サーバ３１が、地上波や衛星回線としてのネットワーク３４を介して、アプリケーションプログラムとしての暗号文をディジタル放送などしている場合には、入力部８１は、その放送されてくる暗号文を受信する。

【００８７】入力部８１は、以上のようにして取得したアプリケーションプログラムとしての暗号文を、そのまま、復号部８２に供給する。

【００８８】復号部８２には、入力部８１から暗号文が供給される他、その暗号文を作成するのに用いた秘密キーに対応する公開キーも供給されるようになされている。

【００８９】ここで、公開キーは、例えば、プログラム認証機関が管理しており、ユーザからの要求に応じて、ネットワーク３４を介して、ユーザ端末３３に送信される。あるいは、公開キーは、郵送などによって、ユーザに配送される。また、プログラムとしての暗号文がファイルとして記録された記録媒体３５がユーザに配布される場合には、公開キーは、記録媒体３５に、暗号文とともに記録しておくようにすることもできる。さらに、公開キーは、プログラム認証機関サーバ３２におけるＪａｖａバイトコードの認証の後に、その認証結果としての暗号文とともに、ソフトウェア開発者サーバ３１に送信するようにし、ソフトウェア開発者から、ユーザに配布するようにすることなども可能である。

【００９０】復号部８２では、公開キーを用いて、入力部８１からの暗号文が復号化され、その復号結果としてのＪａｖａバイトコードが、Ｊａｖａ仮想マシン８３に供給される。Ｊａｖａ仮想マシン８３では、復号部８２からのＪａｖａバイトコードが解釈、実行される。

【００９１】以上のように、復号部８２において、プログラム認証機関サーバ３２での暗号化に用いられた秘密キーに対応する公開キー（秘密キーと対になる公開キー）を用いて、暗号文の復号化が行われ、その復号結果が、Ｊａｖａ仮想マシン８３に入力される。従って、復号部８２に対して、プログラム認証機関で認証されていないＪａｖａバイトコード、即ち、例えば、暗号化されていないＪａｖａバイトコードや、プログラム認証機関サーバ３２における暗号化アルゴリズムと異なるアルゴリズムで暗号化されたＪａｖａバイトコード、あるいはそれと同一のアルゴリズムで暗号化されていたとしても、本来使用すべき秘密キーを用いずに暗号化されたＪａｖａバイトコードなどが入力された場合においては、復号部８２からは、Ｊａｖａ仮想マシン８３が正常に実行可能なＪａｖａバイトコードは出力されないから（Ｊａｖａ仮想マシン８３が正常に実行可能なＪａｖａバイトコードが、偶然に出力されることは有り得ないことではないが、ほとんどないに等しい）、結果として、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードであって、プログラム認証機関で認証されていないものの、Ｊａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布を制限することができる。

【００９２】即ち、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの、そのＪａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Ｊａｖａ仮想マシン８３の開発者や配布者は、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの配布を希望するソフトウェア開発者から、いわば、Ｊａｖａ仮想マシン８３を使用したアプリケーションプログラムを配布するためのライセンス料を得ることが可能となる。

【００９３】なお、Ｊａｖａ仮想マシン８３に対するＪａｖａバイトコードの入力は、復号部８２からのみ行うことができるようにしておく必要がある。

【００９４】ここで、図１２における復号部８２では、何らかの入力があると、その入力に対して復号化処理が施され、その処理結果が出力される。従って、プログラム認証機関で認証されていないＪａｖａバイトコードが復号部８２に入力され、その処理結果が、Ｊａｖａ仮想マシン８３に供給されると、Ｊａｖａ仮想マシン８３は、通常暴走する。そこで、復号部８２の出力が、正当（正常）なＪａｖａバイトコードかどうかを確認し、正当なＪａｖａバイトコードである場合にのみ、そのＪａｖａバイトコードを、Ｊａｖａ仮想マシン８３に解釈、実行させるようにすることが可能である。即ち、Ｊａｖａバイトコードには、その先頭に、マジック（ｍａｇｉｃ）と呼ばれる３２ビットのデータが配置されており、これが本来の値（１６進数で、ＣＡＦＥＢＡＢＥ）である場合にのみ、復号部８２の出力が正当なＪａｖａバイトコードであるとして、Ｊａｖａ仮想マシン８３に解釈、実行させるようにすることができる。この場合、Ｊａｖａ仮想マシン８３の暴走を防止することができる。

【００９５】なお、プログラム認証認証機関サーバ３２において、Ｊａｖａバイトコード全体ではなく、例えば、上述のマジックなどの、Ｊａｖａバイトコードの一部分を暗号化することにより、Ｊａｖａバイトコードの認証を行うことも可能であるが、マジックは３２ビットという少ないビット数で構成されるため、復号部８２において本来の値が出力されるような改竄を、Ｊａｖａバイトコード全体を暗号化する場合に比較して容易に行うことができることが予想される。従って、暗号化は、Ｊａｖａバイトコード全体に対して施すのが望ましい。

【００９６】ところで、図１２の復号部８２における復号化アルゴリズムや、そこで用いるべき公開キーが、第三者に知られても、Ｊａｖａ仮想マシン８３上におけるアプリケーションプログラムの実行の制限の観点からは、暗号化アルゴリズムやその暗号化に用いる秘密キー

さえ知られなければ、特に問題はない。即ち、暗号文の復号化方法が知られても、正常に実行可能なＪａｖａバイトコードをＪａｖａ仮想マシン８３に供給するために復号部８２に与えるべき暗号文の作成方法を知られなければ、仮想マシン８３上におけるアプリケーションプログラムの実行を制限することができる。

【００９７】しかしながら、暗号文の復号化方法が知られた場合、その暗号文（プログラム認証機関においてＪａｖａバイトコードを認証した結果としての暗号文）から、Ｊａｖａバイトコードを得ることができる。Ｊａｖａバイトコードは、前述したように、それを逆コンパイルすることにより、その内容を比較的容易に理解することができるため、リバースエンジニアリングを簡単にすることができる。

【００９８】そこで、このようなリバースエンジニアリングを防止すべく、暗号文の復号化方法は秘密にすることができる。即ち、例えば、暗号文の復号化に用いられる公開キーは、上述したように、一般に公開されるものであるが、これを秘密にすることなどができる。

【００９９】図１３は、公開キーを秘密にするようにしたプログラム実行システムの構成例を示している。なお、図中、図１２における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【０１００】この実施の形態においては、例えば、Ｊａｖａ仮想マシン８３を含むプログラム実行システムを構成するプログラムの中の１カ所に、または複数箇所に分散して、公開キーが配置されており、復号部８２では、その公開キーを用いて、暗号文の復号化が行われる。従って、この場合、公開キーは、プログラム実行システムの外部に漏れることがなく、その結果、暗号文が、不正に復号され、リバースエンジニアリングされることを防止すること（リバースエンジニアリングされる確率を低減すること）が可能となる。

【０１０１】次に、以上においては、ユーザ端末３３において、基本的には、復号部８２における復号化結果を、そのままＪａｖａ仮想マシン８３に入力するようにしたが、ユーザ端末３３においては、Ｊａｖａバイトコードが、プログラム認証機関で認証されたものであり、かつ改竄などが行われていない、いわば正当なものであるかどうかを確認し、その確認がなされたＪａｖａバイトコードのみを、Ｊａｖａ仮想マシン８３に入力するようにすることなども可能である。

【０１０２】この場合、プログラム認証機関サーバ３２では、ソフトウェア開発者サーバ３１から送信されてくるアプリケーションプログラムとしてのＪａｖａバイトコードに対して、例えば、図１４に示すフローチャートにしたがったプログラム認証処理が行われる。

【０１０３】即ち、ステップＳ２１乃至Ｓ２３においては、図１０のステップＳ１１乃至Ｓ１３における場合とそれぞれ同様の処理が行われる。

【０１０４】そして、ステップＳ２２において、送信されてきたＪａｖａバイトコードが、正規のソフトウェア開発者からのものであると判定された場合、ステップＳ２４乃至Ｓ２６に順次進み、そのＪａｖａバイトコードに対して、それが正当なものであることを証明する署名（ディジタルサイン）が付される。

【０１０５】即ち、ステップＳ２４では、ＣＰＵ５１において、Ｊａｖａバイトコードのダイジェストが作成され、ステップＳ２５に進む。ステップＳ２５では、ＣＰＵ５１において、ステップＳ２４で作成されたダイジェストから、ディジタルサインが作成される。そして、ステップＳ２６に進み、通信制御部４７において、そのディジタルサインが、Ｊａｖａバイトコードに付され（このようにディジタルサイン（署名）が付されたＪａｖａバイトコードを、以下、適宜、署名付きバイトコードという）、ソフトウェア開発者サーバ３２に送信されて、処理を終了する。

【０１０６】次に、プログラム認証機関サーバ３２において行われる、ステップＳ２４乃至２６の署名付きバイトコードの作成方法について説明する。

【０１０７】プログラム認証機関サーバ３２では、署名付きバイトコードの作成が、例えば、ＲＳＡ方式などに代表される公開鍵暗号化方式によって行われるようになされている。

【０１０８】即ち、図１５は、公開鍵暗号化方式による、ディジタルサイン（署名）を用いた暗号化／復号化システムの構成例を示している。

【０１０９】ダイジェスト作成器９１には、認証対象である平文が入力される。そして、ダイジェスト作成器９１では、入力された平文のダイジェストが、例えば、ＭＤ５やＳＨＡ－１などのアルゴリズムにしたがって作成される。

【０１１０】ここで、ダイジェストは、平文の機械的な凝縮文に相当し、入力としての平文が異なれば、そのダイジェストも異なるものが作成される。ダイジェストの作成は、平文を、例えば、ハッシュ関数を用いて変換することで行われる。

【０１１１】なお、データベースの検索を行うのに用いるキーワードがとり得る範囲の集合を、ある限られた数値範囲（レコード番号や配列の添字などに対応する）に写像する方法はハッシング（hashing）と呼ばれるが、この写像を行う変換関数がハッシュ関数である。

【０１１２】ダイジェスト作成器９１で作成されたダイジェストは、暗号化器９２に供給される。暗号化器９２では、例えば、図１１の暗号化器７１における場合と同様にして、ダイジェストが、秘密キーを用いて暗号化され、この暗号化されたダイジェストがディジタルサインとして出力される。そして、ディジタルサインが、元の平文に付加され、署名付き平文として出力される。

【０１１３】一方、復号化器９３には、署名付き平文を構成するディジタルサインが、また、ダイジェスト作成器９４には、その残りの平文が、それぞれ入力される。復号化器９３では、例えば、図１１の復号化器７２における場合と同様にして、ディジタルサインが、公開キーを用いて復号化され、ダイジェストとされる。このダイジェストは、署名確認器９５に供給される。

【０１１４】ダイジェスト作成器９４では、ダイジェスト作成器９１における場合と同様にして、そこに入力される平文のダイジェストが作成され、署名確認器９５に供給される。

【０１１５】署名確認器９５では、署名（ディジタルサイン）の正当性が判断される（署名の確認が行われる）。即ち、署名確認器９５では、復号化器９３が出力するダイジェストが、ダイジェスト作成器９４が出力するダイジェストと一致するかどうかが確認される。復号化器９３が出力するダイジェストが、ダイジェスト作成器９４が出力するダイジェストと一致しない場合、例えば、平文の改竄が行われたとして、あるいは、復号化器９３で用いられた公開キーが正しくないとして、署名の正当性が否定される。

【０１１６】一方、復号化器９３が出力するダイジェストが、ダイジェスト作成器９４が出力するダイジェストと一致する場合、平文の改竄が行われておらず、かつ復号化器９３で用いられた公開キーが正しいとして、署名の正当性が確認される。

【０１１７】署名確認器９５には、署名付き平文を構成する平文も供給されるようになされており、そこでは、署名の正当性が確認されると、その平文が出力される。

【０１１８】プログラム認証機関サーバ３２では、ソフトウェア開発者サーバ３１からのＪａｖａバイトコードが、上述の署名付き平文に相当する署名付きバイトコードとされることにより、そのＪａｖａバイトコードが認証される。

【０１１９】なお、署名の手法は、上述した公開鍵暗号化方式に限定されるものではない。

【０１２０】次に、図１６は、ユーザ端末３３においてアプリケーションプログラムの正当性を確認し、正当なもののみを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。なお、図中、図１２おける場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【０１２１】入力部１０１は、基本的には、図１２における入力部８１と同様に、そこへの入力を受け付けるようになされている。但し、入力部１０１には、署名付きバイトコード（署名（ディジタルサイン）が付されたＪａｖａバイトコード）が入力されるようになされており、そこでは、署名付きバイトコードが、署名とＪａｖａバイトコードとに分離されて出力されるようになされている。署名は、署名確認部１０３に、Ｊａｖａバイトコードは、メッセージダイジェストシステム１０２および仮想マシン入力制御部１０４に、それぞれ供給されるようになされている。

【０１２２】メッセージダイジェストシステム１０２は、図１５のダイジェスト作成器９４と同様の処理を行うもので、入力部１０１からのＪａｖａバイトコードからダイジェストを作成し、署名確認部１０３に供給するようになされている。署名確認部１０３は、図１５の復号化器９３および署名確認器９５に相当するもので、入力部１０１からの署名の正当性を確認するようになされている。

【０１２３】即ち、署名確認部１０３には、入力部１０１から署名が、メッセージダイジェストシステム１０２からダイジェストが、それぞれ供給される他、例えば、図１２の復号部８２に公開キーが供給されるのと同様にして、署名を作成するときに用いた秘密キーに対応する公開キーが供給されるようになされている。そして、署名確認部１０３は、その公開キーを用いて、署名を復号化することにより、ダイジェストとし、そのダイジェストと、メッセージダイジェストシステム１０２からのダイジェストとを比較することで、署名の正当性を確認するようになされている。さらに、署名確認部１０３は、その確認結果に対応して、仮想マシン入力制御部１０４を制御するようになされている。

【０１２４】仮想マシン入力制御部１０４は、署名確認部１０３の制御にしたがって、入力部１０１からのＪａｖａバイトコードの、Ｊａｖａ仮想マシン８３への供給を制御するようになされている。

【０１２５】以上のように構成されるプログラム実行システムでは、まず最初に、入力部１０１において、図１２の入力部８１における場合と同様にして、アプリケーションプログラムとしての署名付きバイトコードが取得される。そして、入力部１０１は、その署名付きバイトコードを、署名とＪａｖａバイトコードとに分離し、署名を、署名確認部１０３に、Ｊａｖａバイトコードを、メッセージダイジェストシステム１０２および仮想マシン入力制御部１０４に、それぞれ供給する。

【０１２６】メッセージダイジェストシステム１０２では、入力部１０１からのＪａｖａバイトコードからダイジェストが作成され、署名確認部１０３に供給される。署名確認部１０３では、公開キーを用いて、入力部１０１からの署名が復号化されてダイジェストとされる。さらに、署名確認部１０３では、その復号化したダイジェストが、メッセージダイジェストシステム１０２からのダイジェストと比較され、それが一致するかどうかで、入力部１０１からの署名の正当性が確認される。

【０１２７】署名の正当性が確認された場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム１０２からのダイジェストと一致する場合、

署名確認部１０３は、入力部１０１からのＪａｖａバイトコードを、Ｊａｖａ仮想マシン８３に出力するように、仮想マシン入力制御部１０４を制御する。この場合、仮想マシン入力制御部１０４は、署名確認部１０３の制御にしたがい、入力部１０１からのＪａｖａバイトコードを、Ｊａｖａ仮想マシン８３に供給する。

【０１２８】従って、この場合、Ｊａｖａ仮想マシン８３では、入力部１０１から仮想マシン入力制御部１０４を介して供給されるＪａｖａバイトコードが解釈、実行される。

【０１２９】一方、署名の正当性が確認されなかった場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム１０２からのダイジェストと一致しない場合、署名確認部１０３は、入力部１０１からのＪａｖａバイトコードを、Ｊａｖａ仮想マシン８３に出力しないように、仮想マシン入力制御部１０４を制御する。

【０１３０】この場合、仮想マシン入力制御部１０４からＪａｖａ仮想マシン８３に対しては、入力部１０１からのＪａｖａバイトコードは出力されず、従って、Ｊａｖａ仮想マシン８３では、特に処理は行われない。

【０１３１】以上から、アプリケーションプログラムの認証として署名を付す場合も、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードであって、プログラム認証機関で認証されていないものの、Ｊａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布を制限することができる。即ち、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの、そのＪａｖａ仮想マシン８３が実装されたユーザ端末３３を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Ｊａｖａ仮想マシン８３の開発者等は、Ｊａｖａ仮想マシン８３上で動作するＪａｖａバイトコードの配布を希望するソフトウェア開発者から、ライセンス料を得ることが可能となる。

【０１３２】また、署名を付す場合においては、その署名を付したＪａｖａバイトコードを改竄したものの、仮想マシン８３上での実行も制限することができる。

【０１３３】なお、図１６の実施の形態では、Ｊａｖａ仮想マシン８３に対するＪａｖａバイトコードの入力は、仮想マシン入力制御部１０４からのみ行うことができるようにしておく必要がある。

【０１３４】ここで、Ｊａｖａバイトコードに署名を付す場合においては、Ｊａｖａバイトコードを暗号化する場合と異なり、Ｊａｖａバイトコードそのものが存在する。従って、プログラム実行システムが署名の正当性を確認しないもの（例えば、入力部１０１の出力であるＪａｖａバイトコードが、Ｊａｖａ仮想マシン８３に直接入力されるような構成のもの）であれば、そのプログラム実行システムでは、何の制限もなく、Ｊａｖａバイトコードを解釈して実行することができる。

【０１３５】即ち、逆にいえば、Ｊａｖａバイトコードに署名を付す場合においては、その実行を制限したいＪａｖａ仮想マシンの開発者や販売者、さらには、ユーザ端末３３にＪａｖａ仮想マシンを実装して販売する者などは、そのプログラム実行システムを、図１６に示したように構成すれば良いし、その実行の制限を特に希望しない者は、プログラム実行システムを署名の正当性を確認しないような構成とすれば良い。

【０１３６】なお、本発明は、上述したインタープリタ形式およびＪＩＴコンパイラ形式のいずれのＪａｖａ仮想マシンにも適用可能であるし、また、Ｊａｖａ仮想マシン以外の仮想マシン、さらには、例えば、Ｃ言語や、Ｃ＋＋言語などの処理系のようにプログラム実行システムへの入力が機械語コードの場合や、Ｂａｓｉｃ言語の処理系のようにプログラム実行システムへの入力がソースコードの場合などについても適用可能である。

【０１３７】また、図１２や図１３、図１６の実施の形態においては、プログラム実行システムを１つしか設けていないが、ユーザ端末３３には、このプログラム実行システムを複数設けることも可能である。例えば、入力部８１や１０１を複数設けた場合においては、複数経路からの暗号文や署名付きバイトコードの入力が可能となる。さらに、例えば、復号部８２を複数設けた場合においては、複数の復号アルゴリズムにしたがって暗号文を復号することが可能となる。また、例えば、Ｊａｖａ仮想マシン８３を複数設けた場合には、複数のＪａｖａバイトコード形式をサポートすることが可能となる。さらに、例えば、メッセージダイジェストシステム１０２および署名確認部１０３を複数設けた場合には、複数の手法それぞれで付された複数の署名の確認を行うことが可能となる。

【０１３８】なお、本実施の形態におけるＪａｖａバイトコードとは、例えば、Java Application, Java Applet, Java Benas, Java Class Libraryその他の多数の形態のＪａｖａバイトコードをすべて含むものである。

【０１３９】

【発明の効果】請求項１に記載の情報処理装置および請求項２に記載の情報処理方法によれば、暗号化されたプログラムが復号化され、その復号化されたプログラムが実行される。また、請求項３に記載の記録媒体には、コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。従って、暗号化されたプログラムのみが実行されるようにすることが可能となる。

【０１４０】請求項５に記載の情報処理装置および請求項６に記載の情報処理方法によれば、請求項１に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムが暗号化される。また、請求項７に記載の記録媒体には、請求項１に記載の情報処理装置で実行

可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されている。従って、請求項１に記載の情報処理装置で実行可能な、暗号化されたプログラムの提供が可能となる。

【０１４１】請求項８に記載の情報処理装置および請求項１０に記載の情報処理方法によれば、プログラムが正当なものかどうかが確認され、正当なものであることが確認された場合のみ、プログラムが実行される。また、請求項１１に記載の記録媒体には、コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されている。従って、正当なプログラムのみが実行されるようにすることが可能となる。

【０１４２】請求項１２に記載の情報処理装置および請求項１３に記載の情報処理方法によれば、請求項８に記載の情報処理装置において正当なものであると確認されるように、プログラムが処理される。また、請求項１４に記載の記録媒体には、請求項８に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されている。従って、請求項８に記載の情報処理装置で実行可能なように処理されたプログラムの提供が可能となる。

【図面の簡単な説明】

【図１】計算機１の資源と、そこに実装されたＪａｖａ仮想マシン１１の資源との対応関係を示す図である。

【図２】Ｊａｖａ仮想マシン１１の処理を説明するための図である。

【図３】Ｊａｖａ仮想マシン１１の処理を説明するための図である。

【図４】Ｊａｖａ仮想マシン１１の処理を説明するための図である。

【図５】本発明を適用したプログラム提供システムの一実施の構成例を示すブロック図である。

【図６】図５のソフトウェア開発者サーバ３１の構成例を示すブロック図である。

【図７】図５のプログラム認証機関サーバ３２の構成例を示すブロック図である。

【図８】図５のユーザ端末３３の構成例を示すブロック図である。

【図９】ソフトウェア開発者サーバ３１の処理を説明するためのフローチャートである。

【図１０】プログラム認証機関サーバ３２の処理を説明するためのフローチャートである。

【図１１】暗号化／復号化システムの構成例を示すブロック図である。

【図１２】プログラム実行システムの第１の機能的構成例を示すブロック図である。

【図１３】プログラム実行システムの第２の機能的構成例を示すブロック図である。

【図１４】プログラム認証機関サーバ３２の処理を説明するためのフローチャートである。

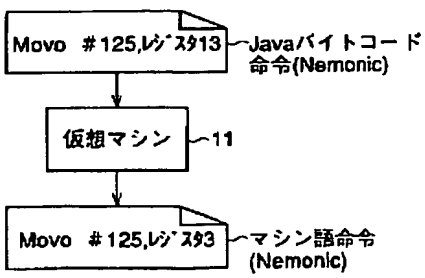
【図１５】ディジタルサインを用いた暗号化／復号化システムの構成例を示すブロック図である。

【図１６】プログラム実行システムの第３の機能的構成例を示すブロック図である。

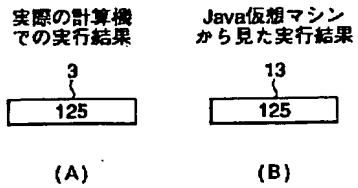
【符号の説明】

１　計算機，　２　中央演算処理装置，　３　レジスタ，　４　メモリ，　１１　Ｊａｖａ仮想マシン，　１３　レジスタ，　１４　メモリ，　２１　Ｊａｖａコンパイラ，　３１　ソフトウェア開発者サーバ，　３２　プログラム認証機関サーバ，　３３　ユーザ端末，　３４　ネットワーク，　３５　記録媒体，４１　ＣＰＵ，　４２　ＲＯＭ，　４３　ＲＡＭ，　４４　入力部，　４５　出力部，　４６　補助記憶装置，　４７　通信制御部，　５１　ＣＰＵ，　５２ＲＯＭ，　５３　ＲＡＭ，　５４　入力部，　５５　出力部，　５６　補助記憶装置，　５７　通信制御部，　６１　ＣＰＵ，　６２　ＲＯＭ，　６３　ＲＡＭ，　６４　入力部，　６５　出力部，　６６　補助記憶装置，　６７　通信制御部，　７１　暗号化器，　７２　復号化器，　８１　入力部，　８２　復号部，８３　Ｊａｖａ仮想マシン，　９１　ダイジェスト作成器，　９２　暗号化器，　９３　復号化器，　９４　ダイジェスト作成器，　９５　署名確認器，　１０１　入力部，　１０２　メッセージダイジェストシステム，　１０３　署名確認部，　１０４　仮想マシン入力制御部

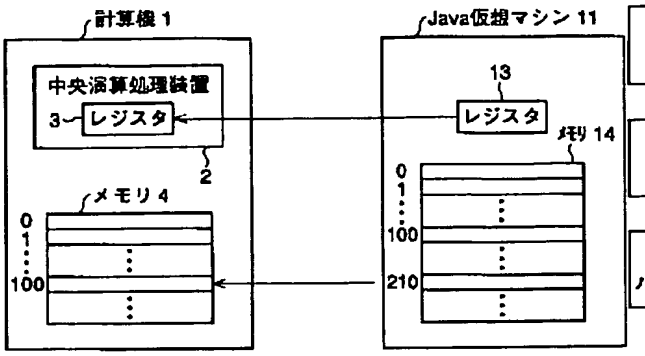
【図３】



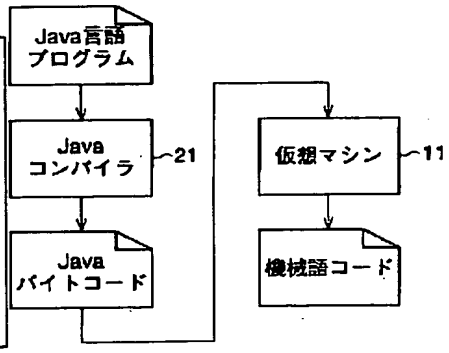
【図４】



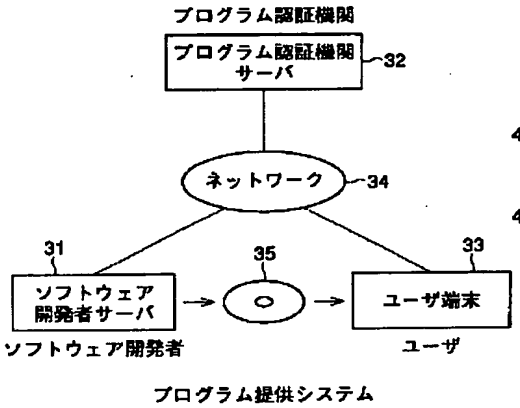
【図１】



【図２】

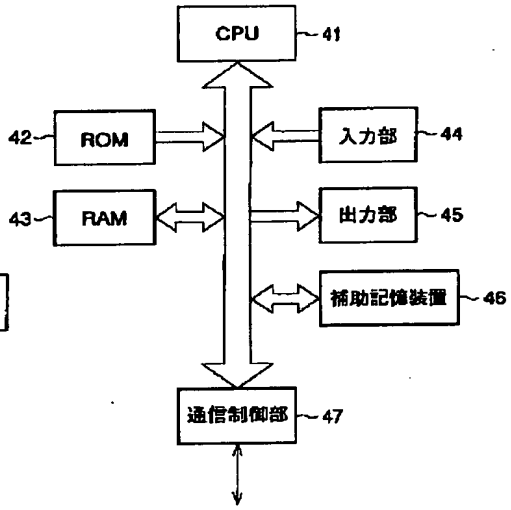


【図５】



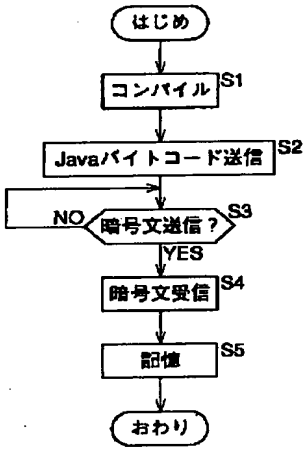
プログラム提供システム

【図６】

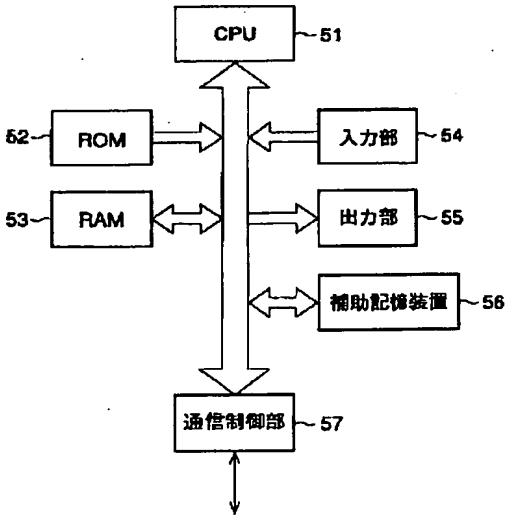


ソフトウェア開発者サーバ 31

【図９】

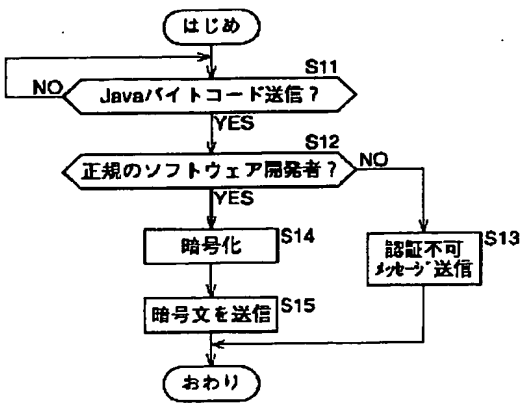


【図７】

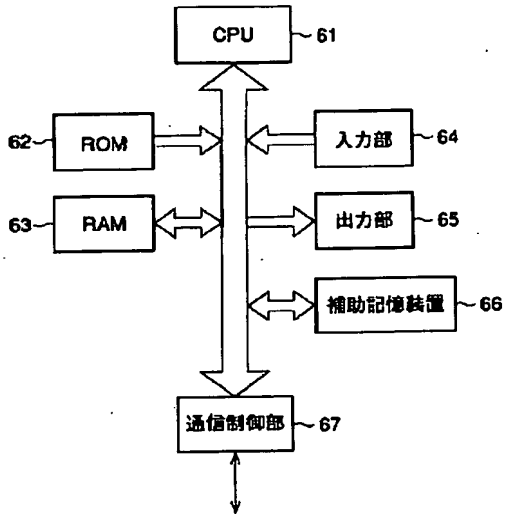


プログラム認証機関サーバ 32

【図１０】

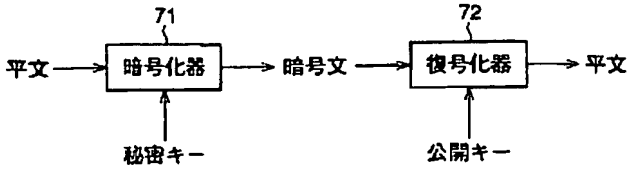


【図８】



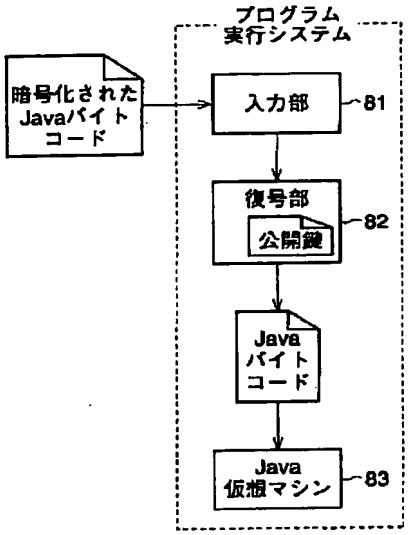
ユーザ端末 33

【図１１】

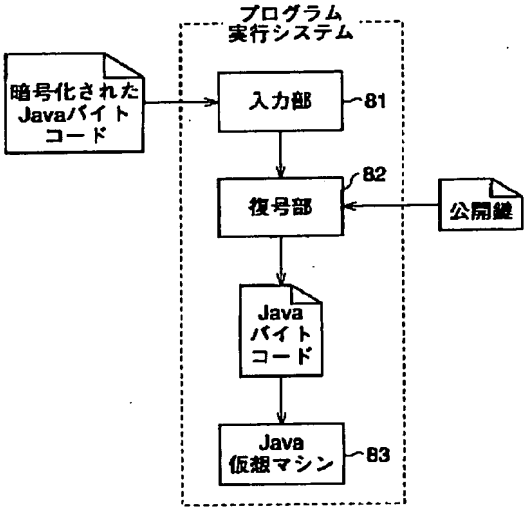


公開キー暗号化／復号化システム

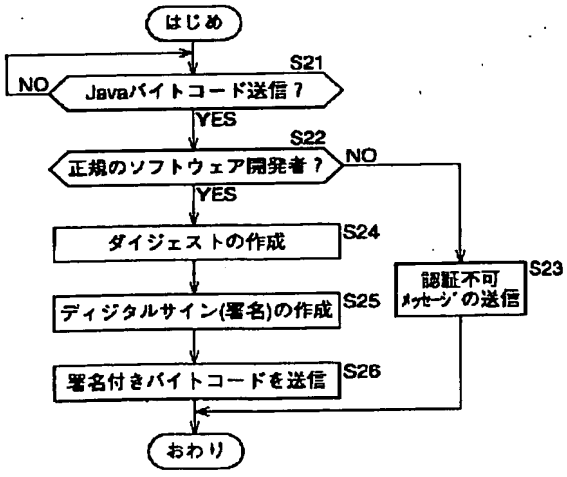
【図１３】



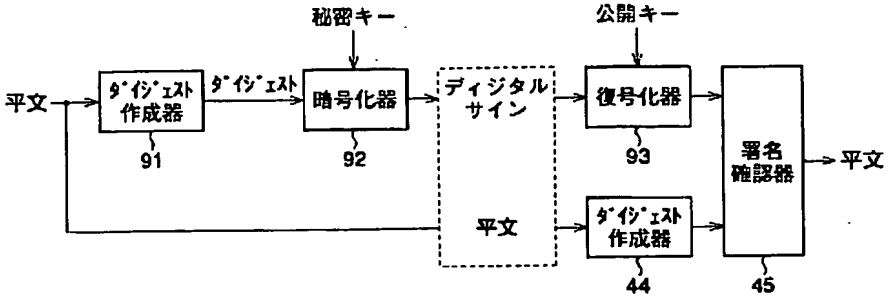
【図１２】



【図１４】



【図１５】



ディジタルサインを用いた暗号化／復号化システム

【図１６】